AMENDMENTS TO THE SPECIFICATION

Please amend the specification as indicated hereafter. It is believed that the following amendments and additions add no new matter to the present application.

In the Specification: [Use ~~strikethrough~~ for deleted matter (or double square brackets "[[]]" if the strikethrough is not easily perceivable, *i.e.*, "4" or a punctuation mark) and underlined for added matter.]

Please amend the paragraph starting on p. 3, line 20 as follows:

~~In a preferred embodiment, the method deriving synthesizing functions which map measurement responses of the circuit to circuit performance parameters, applying an optimized input stimulus to the circuit, capturing the circuit response to the input stimulus applied to the circuit, evaluating the circuit response with respect to the derived synthesizing functions to predict whether a predetermined number of performance parameters of the circuit satisfies predetermined specifications for the circuit, making a pass/fail determination for the circuit based upon the evaluation of the circuit response, and for circuits for which a clear pass/fail determination cannot be made, performing specification-based tests with respect to particular predetermined circuit specifications to make a final pass/fail determination for the circuit.~~

Please amend the paragraph starting on p. 5, line 7 as follows:

FIG. 8 is a ~~is a~~ schematic of a routine showing steps involved in synthesizing mapping functions.

Please amend the paragraph starting on p. 23, line 3 as follows:

For efficient production testing of analog circuits, it is desirable to minimize the number of measurements. By minimizing the number of measurements, the variance, σ_{emi}^2, is minimized. Moreover, the average production testing time is also minimized. Nevertheless, eliminating measurements might ~~led~~ lead to loss of information about the performance parameters of the circuit~~. This~~; thus an increase in the variance σ_{eri}^2. Hence, only those measurements should be eliminated if used for deriving the synthesized measurements increases the overall variance σ_{ei}^2.

Please amend the paragraph starting on p. 23, line 16 as follows:

The procedure OrderMeasurements, shown in FIG. 8, removes one measurement at a time from the list of measurements and calculates the variance σ_{ei}^2 of the synthesizing function derived using the remaining measurements. The measurements are then ordered in the ascending order of the variance σ_{ei}^2. The procedure SelectMeasurements (FIG. 8) then takes this ordered list of measurements and adds one measurement at a time to the list of selected measurements and derives the synthesizing function using these selected measurements. Initially, the overall variance σ_{ei}^2 reduces because we increase the accuracy of our regression by adding more measurements. After adding a few measurements, the addition of more measurements ~~do~~ does not result in an increase in the accuracy of the regression model. However, the variance σ_{emi}^2 increases because we are introducing more measurement error with the addition of a new measurement. This leads to an increase in the overall variance σ_{ei}^2. Once σ_{ei}^2 start increasing, we stop adding new measurements and the final list of measurements will give a minimum variance of the errors.

Please amend the paragraph starting on p. 25, line 2 as follows:

As explained above, the chief reasons for misclassification during testing are the measurement errors and regression errors. The problem of misclassification becomes severe near the boundaries of the acceptance region. This can be explained with the help of Figure 9. In the figure, σ_p is the standard deviation of the measurements around the nominal fault free circuit instance due to the process ~~vaiations~~ variations and σ_e is that due to the measurement errors. Circuit instance p_1 is well within the region of acceptance, p_3 is outside the acceptance region and p_2 is at the measurement threshold. In the figure we show the distribution of the measurements performed on these circuit instances. From the figure it is clear that p_1 and p_3 are always classified correctly even in the presence of measurement errors. However, there is a high probability that, p_2, the circuit instance at the boundary of the acceptance region is misclassified. The chances of misclassification reduces if the ratio is σ_p/σ_e increased. Thus a good test ~~need~~ needs to have the standard deviation due to the process variations much greater than that due to the measurements. An objective during test design is to increase the standard deviation σ_p by appropriately choosing the measurements. This, in turn, can be achieved by selecting those measurements which are sensitive to the process parameter deviations of the circuit.

Please amend the paragraph starting on p. 25, line 18 as follows:

Earlier sensitivity based test generation methods were formulized to maximize the sensitivity of the measurements to the process parameter variations of the nominal circuit instance. Maximizing the sensitivity of the measurements to the process parameter deviations around the nominal fault free circuit often does not help, especially for circuits with loose specifications. If the specifications are not tight, increasing the sensitivity of the

measurements to the process parameters for the nominal circuit will not necessarily increase the sensitivity of the measurements to the process parameters for the circuit instances near the boundary. This will lead to large rates of misclassification for a drifted manufacturing process. Thus our objective during test generation is to derive measurements which are highly sensitive to the process deviations of the critical circuit instances. Thus the fitness function for a set of n_m measurements is given by equation [23], where n_c is the number of critical circuit instances. We select one worst-case circuit instance per single-ended specification to evaluate the fitness. This worst-case critical circuit instance is the one which is nearest to the nominal circuit instance (lowest l_2 norm) among all those generated by GenerateTrainSet()

$$Fitness = \sum_{j=1}^{n_m} \sum_{i=1}^{n_p} \sum_{k=1}^{n_c} \left| (S_{p_i}^{m_j})_k \right| \qquad \text{[Equation 21]}$$

based on their fitness. After crossover and mutation of the existing population, the evolved new population again undergoes selection, crossover and mutation to give a population with individuals having better fitness functions. To conduct the genetic search, the search space typically must be encoded into genetic strings or chromosomes. A [[a]] set of rules must be provided for selection, crossover, mutation and fitness evaluation for these genetic strings. For example, the following rules could be established:

String encoding: The i^{th} gene of the genetic string is an integer representing the voltage at time point t_i (equation [22]). Thus, if the i^{th} gene has a value j, the corresponding voltage of the PWL transient waveform is given by equation [23]. The string length of the population is equal to the total number of time divisions n_t. FIG. 10 shows the encoding in

detail. In the figure, maximum voltage was assumed to be 5V and there are 5 voltage divisions and 5 time divisions.

$$t_i = \frac{t_{\max}}{n_i} i \qquad \text{[Equation 22]}$$

$$Vin(t_i) = \frac{Vin_{\max}}{n_v} j \qquad \text{[Equation 23]}$$

<u>Selection</u>: The selection of strings for crossover is biased towards strings having the highest fitness value so that the average fitness of successive populations tends to increase. Tournament selection can be used for selecting the parents for reproduction. Tournament selection involves picking two strings from the population and selecting the better for reproduction.

<u>Crossover</u>: The crossover operator takes genes from each of the parent ~~string~~ <u>strings</u> and combines them to create child strings. A uniform crossover scheme can then be used for creating child strings. FIG. 11 shows how uniform crossover is performed to produce the child strings. Each gene of the parent strings is chosen with certain probability and are swapped to yield the two child strings.

<u>Mutation</u>: After the child strings are created, the genes of the child strings can undergo mutation. For mutation, a gene is selected with a certain probability (mutation probability) and is replaced with a random number within the allowed range.

<u>Fitness evaluation</u>: One measurement in a particular segment of the PWL transient waveform is suade. Hence, he total number of measurements is equal to the string length N. The fitness is equal to the sum of absolute value of the sensitivities of the measurements to the process parameters for the critical circuit instances.

Please amend the paragraph starting on p. 30, line 3 as follows:

Out of 1000 circuits, 776 were found to satisfy the specification on bias current. Using the synthesized measurements 744 of these circuits were classified with a high degree of certainty. Out of the remaining 32 circuits, 17 had an uncertain test outcome. Hence for these 17 circuits, bias current is a critical performance parameter. Similarly, 20 circuit instances (out of 224) ~~those~~ that did not satisfy the specification on the bias current have bias current as a critical performance parameter.

Please amend the paragraph starting on p. 30, line 9 as follows:

When all the specifications are considered, the synthesized measurements are able to correctly classify 93% of the CUTs. Out of the remaining 7%, 4.5% had an uncertain test outcome ~~stet~~ stat. These circuits are then subjected to the critical specification tests for fault detection. Thus only remaining 2.5% of the CUTs are misclassified by the proposed test procedure.